

CLAIMS

Please amend the claims as follows:

Claim 1 (currently amended) A computer-implemented method using a computer having a display and a memory for storage of information representative of hydrologic and hydraulic conditions for determining water flow directions based on flow vector[[s]] values calculated in part with a D8 method applied to all or part of a watershed divided into more than one predefined, rectangular major cell each encompassing at least nine predefined, rectangular minor cells including a center minor cell and eight neighbor minor cells wherein the watershed includes one or more types of problem areas identified as depressions, flat areas or peaks each of which extends across one or more than one major cell[[s]] and each of which requires fixing and wherein further the number, location and elevation of all minor cells are known as are a maximum depth depression to be corrected, a maximum number of downstream minor cells to check in fixing depressions and peaks, a largest depression area to fix and a largest flat area to fix, comprising:

~~if flow vectors for minor cells within more than one major cell are desired,~~
selecting and designating one major cell for which to begin generating flow vector[[s]] values;

creating a buffered array in the memory containing elevation and identification data for all minor cells in the designated major cell and all major cells, if any, bordering the designated major cell;

calculating flow vector values for each minor cell within the buffered array using the D8 method wherein a flow vector value of zero is assigned to any minor cell located in a problem area for which a flow vector value cannot be calculated using the D8 method;

storing in the memory the calculated flow vector values in association with the minor cell to which they relate;

checking the flow vector value of each minor cell within the designated major cell until all cells have been checked and all cells having a flow vector value of zero have been found;

assigning a problem type in the memory to each minor cell having a flow vector value of zero;

fixing first all minor cells in the designated major cell to which a depression problem type has been assigned wherein depressions are fixed independently and successively from the highest depression to the lowest depression and elevation changes are imposed on minor cells, as required;

fixing second all minor cells in the designated major cell to which a flat area problem type has been assigned wherein each flat area is fixed independently and elevation changes are imposed on minor cells, as required;

fixing third all minor cells in the designated major cell to which a peak problem type has been assigned wherein peaks are fixed independently and successively from the lowest peak to the highest peak and elevation changes are imposed on minor cells, as required;

recalculating and storing in the memory flow vector values for each minor cell within the buffered array using the D8 method wherein a flow vector value of zero is assigned to any minor cell located in a problem area for which a flow vector value cannot be calculated using the D8 method;

checking again the flow vector value of each minor cell within the designated major cell until all cells have been checked and all cells having a flow vector value of zero, if any, are found;

if any minor cells having a flow vector value of zero have been found,

assigning a problem type in the memory to each minor cell having a flow vector value of zero;

fixing first all minor cells in the designated major cell to which a depression problem type has been assigned wherein depressions are fixed independently and successively from the highest depression to the lowest depression by imposing elevation changes on minor cells, as required;

fixing second all minor cells in the designated major cell to which a flat area problem type has been assigned wherein each flat area is fixed independently by imposing elevation changes on minor cells, as required;

fixing third all minor cells in the designated major cell to which a peak problem type has been assigned wherein peaks are fixed independently and successively

from the lowest peak to the highest peak and elevation changes are imposed on minor cells, as required;

storing in the memory the elevation and flow vector [[data]] value associated with each minor cell as changed, if necessary;

selecting and designating a previously unselected major cell and repeating the process beginning with creating a buffered array; and

~~if there are unselected major cells remaining, selecting another major cell, and returning to the creating a buffered array step; and~~

retrieving the elevation and flow vector value for the watershed stored in the memory to depict on the display at least one of a basin delineation, time of concentration calculation, flowpath tracing, drainage area calculation, the extents of hazardous waste spills and a streamline.

Claim 2 (currently amended) The method of claim 1 wherein [[the]] calculating [[step]] for each minor cell further comprises:

retrieving identification and elevation data for the minor cell;

if that minor cell is located on the edge of the buffered array, assigning a flow vector value of zero to that minor cell;

if a distinct flow direction from that minor cell to one of its eight neighboring minor cells can be ascertained, assigning a flow vector value of between one and nine to the minor cell wherein each number represents a different direction; and

otherwise assigning a flow vector value of zero to the minor cell.

Claim 3 (currently amended) The method of claim 1 wherein [[the]] assigning [[step]] for each minor cell having a zero flow vector value further comprises:

obtaining the location and elevation of all neighbor minor cells of the center minor cell;

if the minor cell is at the edge of the buffered array, assigning the problem type as “edge”;

calculating the slope from the minor cell to each of its eight, surrounding neighbor cells in order to find a maximum slope and,

if the elevation of the minor cell is lower than each of those neighbor minor cells, assigning the problem type as “depression”;

if the maximum slope from the center minor cell to one or more neighbor minor cells equals zero assigning the problem type as “flat”; and

if the maximum slope from the center minor cell to more than one neighbor minor cell does not equal zero, assigning the problem type as “peak”.

Claim 4 (currently amended) The method of claim 1 wherein fixing first is a subroutine comprising: A computer implemented method using a computer having a display and a memory for storage of information representative of hydrologic and hydraulic conditions for determining water flow directions based on flow vectors calculated in part with a D8 method applied to all or part of a watershed divided into more than one predefined, rectangular major cell each encompassing at least nine predefined, rectangular minor cells including a center minor cell and eight neighbor minor cells, wherein the watershed includes one or more problem areas identified as small spurious depressions or large, man-made depressions requiring flow vector correction of minor cells therein, each of which extends across one or more major cells, wherein the number, location and elevation of all minor cells are known as are a largest depression area to be corrected, a maximum number of times a maximum depth depression may be modified in an attempt to fix any individual depression and the maximum number of downstream cells greater than two permitted to be checked to find a good outlet and wherein further a buffered array containing elevation and identification data for all minor cells in a designated major cell and in all major cells bordering the designated major cell has been established in the memory and flow vector values for all minor cells in the buffered array have been calculated and stored in the memory to the extent possible based on a D8 method and any minor cell located in a depression has been assigned a zero flow vector value, comprising:

initializing a first variable in the memory to track the highest depression elevation found;

flagging in the memory that original minor cell in the designated major cell having a zero vector value which has the highest elevation as compared with all other minor cells having zero vector values located in the designated major cell;

retrieving from the memory location and elevation data for the flagged original minor cell and its neighbor minor cells;

setting a depth limit in the memory to a preselected amount to limit the maximum depth depression to correct;

initializing a second variable in the memory to track the number of times the maximum depth depression has been modified up to a preselected maximum number of times;

creating a depression array in the memory mirroring the buffered array for identifying those minor cells which contribute to the depression;

increasing the second variable in the memory by one;

defining the area contributing to the depression;

identifying potential outlets from the depression;

determining whether any of the potential outlets from the depression is a good outlet;

if no good outlet is found,

 exiting the method subroutine if

 the contributing area extends beyond the buffered array; or

 the number of maximum depth modification attempts equals the preselected maximum number of times;

 otherwise, incrementing the depth limit in the memory by an amount equal to one-eighth of the depth limit;

 if the depth limit is greater than the combination of the elevation of the flagged original minor cell and the depth limit as originally set, exiting the method subroutine;

 otherwise, returning to [[the]] creating a depression array [[step]];

 if a good outlet exists,

 choosing the good outlet having the lowest elevation;

drawing a hypothetical straight line in the memory from the center of the flagged original minor cell to the good outlet cell;

extending the hypothetical straight line in the memory to follow a downstream path to an elevation lower than the elevation of the flagged original minor cell;

burning a cutline into the buffered array in the memory by correcting elevations in minor cells, as necessary;

recalculating and storing in the memory the flow vector[[s]] values for all minor cells in the buffered array if the depression is fixed; and

until all depressions are fixed or there is no more data, returning to [[the]] flagging [[step]].

Claim 5 - 7 (canceled).

Claim 8 (original) The method of claim 4 wherein the depth limit is initially set at one-half of the originally specified maximum depth depression.

Claim 9 (currently amended) The method of claim 4 wherein [[the]] defining [[step]] further comprises:

obtaining data identifying the location and elevation of a center minor cell; testing each of the eight neighbor minor cells surrounding the center minor cell to determine if the elevation of any neighbor minor cell is greater than or equal to the elevation of the center minor cell;

if so, flagging the center minor cell as a contributing minor cell and redesignating that neighbor minor cell as the center minor cell if the elevation of the neighbor minor cell is less than the depth limit and the neighbor minor cell is not located at the edge of the buffered array;

returning to [[the]] obtaining data [[step]] until all neighbor minor cells of all minor cells in the buffered array have been tested or until a stack space error is encountered or the center minor cell is determined to be located at the edge of the buffered array in which case the method subroutine is exited.

Claim 10 (currently amended) The method of claim 9 including the following after [[the]] defining [[step]]:

if an error is encountered during the defining,

exiting the method subroutine if stack space has been exceeded, the area has already been flagged as too shallow or the preselected maximum number of times to attempt fixing the depression array has been reached;

incrementing the depth limit;

if the depth limit after incrementing is not less than the original elevation,

resetting the depression array;

incrementing the second variable by one; and

returning to [[the]] defining [[step]];

exiting the method subroutine if the depth limit after incrementing is less than the original elevation.

Claim 11 (currently amended) The method of claim 10 wherein in [[the]] incrementing [[step]] the depth limit is increased by one-eighth of the initial depth limit.

Claim 12 (currently amended) The method of claim 4, wherein [[the]] identifying potential outlets [[step]] further comprises:

obtaining data identifying the location and elevation of a contributing minor cell and of its neighbor minor cells ;

identifying which of the eight neighbor minor cells of the contributing minor cell is not part of the contributing area;

testing whether the elevation of any such neighbor minor cell is less than the elevation of the contributing minor cell;

if so, flagging such neighbor minor cell as an exterior cell;

after all neighbor minor cells have been so tested, further flagging the contributing minor cell as a potential outlet if at least one neighbor minor cell has been flagged as an exterior cell.

Claim 13 (currently amended) The method of claim 12, wherein determining is a subroutine further comprising:

obtaining data identifying the location and elevation of a contributing minor cell flagged as a potential outlet and of its neighbor minor cells;

setting a trace counter equal to zero;

incrementing the trace counter by one;

exiting the subroutine if the trace counter indicates a value greater than the maximum number of downstream cells permitted to be checked;

obtaining data for the location and elevation of the next downstream minor cell;

testing whether that downstream cell is located at the edge of the buffered array;

if so, exiting the subroutine;

if not, further testing whether the elevation of the downstream minor cell is less than the elevation of the flagged original minor cell;

if so, flagging the potential outlet as a good outlet and exiting the subroutine; and

if not, returning to [[the]] incrementing[[step]].

Claim 14 (currently amended) The method of claim 1 wherein fixing third is a subroutine comprising: A computer implemented method using a computer having a memory for storage of information representative of hydrologic and hydraulic conditions for determining water flow directions based on flow vectors calculated in part with a D8 method applied to all or part of a watershed divided into more than one predefined, rectangular major cell encompassing at least nine predefined, rectangular minor cells including a center minor cell and eight neighbor minor cells, wherein the watershed includes one or more problem areas identified as peaks requiring flow vector correction of minor cells therein, each of which extends across one or more one major cells, and the number, location and elevation of all minor cells are known as are a maximum number of downstream cells greater than two to check and wherein further a buffered array containing elevation and identification data for all minor cells in a designated major cell and in all major cells, if any, bordering the designated major cell has been established in the memory and flow vector values for all minor cells in the buffered array have been calculated and

~~stored in the memory to the extent possible based on a D8 method and any minor cell located in a peak has been assigned a zero flow vector value, comprising:~~

flagging in the memory that original minor cell in the designated major cell having a zero vector value which has the lowest elevation as compared with all other minor cells having zero vector values located in the designated major cell;

retrieving from the memory the location and elevation data for the flagged original minor cell and its neighbor minor cells;

finding and storing in the memory the maximum slope existing between the flagged original minor cell and one of its neighbor minor cells using the D8 method;

identifying each neighbor minor cell where the slope between that cell and the flagged original minor cell equals the maximum slope;

marking each such neighbor minor cell in the memory as a potential path;

if there are not multiple potential paths, exiting the ~~method subroutine~~;

tracing each potential path downstream from each flagged neighbor minor cell up to the specified maximum number of downstream cell iterations wherein for each iteration the maximum slope is reset in the memory to be equal to the steepest slope between a downstream cell and one of its neighbor minor cells if that slope is the steepest slope found from that downstream cell;

selecting a single potential path at random from all of the initial possible potential paths if either there are no potential paths remaining to be traced or if there are multiple potential paths remaining to be traced;

designating the remaining single potential path in the memory as the best path;

lowering the elevation stored in the memory for the first neighboring minor cell of the flagged original minor cell along which the best path runs;

recalculating and storing in the memory flow vector[[s]] values for all minor cells in the buffered array if the peak is fixed; and

until all peaks are fixed or there is no more data, returning to [[the]] flagging [[step]].

Claim 15 - 16 (canceled).

Claim 17 (currently amended) The method of claim 14 wherein ~~randomizing~~
selecting a single potential path further comprises:

identifying each potential flow path with a sequential number beginning at one;
dividing the number of remaining potential flow paths by two to obtain a quotient;
and
designating the sequential number of the potential flow path corresponding to the
quotient as the best flow path.

Claim 18 (original) The method of claim 14 wherein the lowering step comprises
reducing the elevation by .000001 feet.

Claim 19 (currently amended) The method of claim 14 wherein ~~the following~~
~~further steps are included after the lowering~~ [[step]] is followed by:

verifying that lowering of the elevation does not introduce unexpected problems or
undesirable flow direction changes; and
if problems or changes were introduced, returning the elevation to its original level
and exiting the ~~method~~ subroutine.